محمد آرمانلو ۸۱۰۱۰۰۰۸۴ تکلیف سوم نظریه زبان استاد حجت

* یک روز دیر

(۱)

a) $L_1 = \{0^i 1^j \mid \gcd(i,j) = 1\}$

Demon picks $p \geq 1$

you pick $0^{p+1} 1^p$ $\quad (\gcd(i, i+1) = 1)$ ✓

Demon : $w = xyz$ $\quad x = 0^l$ $\quad y = 0^j$ $\quad z = 0^{1+p-p'} 1^p$

you pick $i = \frac{p-1}{j} + 1$ $\quad 0^{l + ij + 1 + p - p'} 1^p$

چگونه بدست آمدن i

$$\boxed{= 0^{(i-1)j+1+p} 1^p \rightarrow (i-1)j + 1 \overset{p}{\equiv} 0 \rightarrow (i-1)j = p-1 \rightarrow i = \boxed{\frac{p-1}{j} + 1}}$$

با قرار دادن i برابر مقدار بالا ما برنده بازی میشویم پس زبان نامنظم است

b) $L_2 = \{(ab)^n a^k \mid n > k,\ k \geq 0\}$

Demon picks $p \geq 1$

you pick $(ab)^{p+1} a^p$

$w = xyz$ $\quad |xy| = p' \leq p$ and $y \neq \varepsilon$

$x = (ab)^l$ $\quad y = (ab)^j$ $\quad z = (ab)^{p-p'+1} a^p$

take $i = 0$ $\rightarrow$ $w = xz = (ab)^{p+1-j} a^p$

$(j \geq 1)$ $\quad p+1-j \leq p$ $\quad$ you win! ✓

so L_2 is not a regular language

c) $L_3 = \{w \mid w \in \{a,b,c\}^* \text{ and } n_a(w) \leq n_b(w) \leq n_c(w)\}$

Demon picks $p \geq 1$

You picks $c^p b^p a^p$

Demon $w = xyz$ $\quad x = c^l \quad y = c^j \quad z = c^{p-p'} b^p a^p$

You pick $i = 0$ $\rightarrow w' = c^l c^{p-p'} b^p a^p$

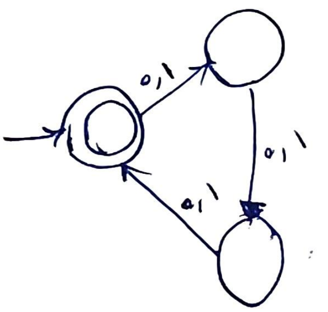
$p' = l + j \quad \Rightarrow \quad l + p - p' < p$ ✗

$j \geq 1$

You win the game so L_3 is not a regular languege

۲

a) we can write DFA



if $|w| \overset{3}{\equiv} 0$, we can divide each letter into two groups and make a group twice larger than other

also we cant use less than 3 states for DFA

b) Demon pick $p \geq 1$

You pick $a^p b^p c^{p+1}$

Demon $w = xyz$ $\quad x = a^l \quad y = a^j \quad z = a^{p-p'} b^p c^{p+1}$

You pick $i = 0$ $\rightarrow$ we know, $a^{l+p-p'} b^p c^{p+1} \notin L$

because $l+p-p' = l+p-l-j = p-j \neq p$

c) Demon picks $p \geq 1$

You picks $w = a^{Prime}$ $\quad Prime > P$

Demon picks $w = xyz$ $\quad x = a^{l}$ $\quad y = a^{j}$ $\quad z = a^{P-P'}$

You pick $i = \frac{Prime}{j} + 1$

$$\rightarrow w' = a^{l + ji + P - P'} = a^{Prime + (i-1)j}$$

$$\rightarrow Prime + (i-1)j = 2 \times prime \rightarrow (i-1) = \frac{prime}{j}$$

$$\Rightarrow i = \frac{prime}{j} + 1$$

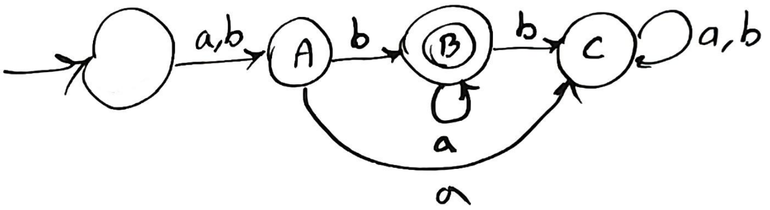
if we set $i = \frac{Prime}{j} + 1$ we win so L_3 isn't regular

---

۳

a)

	a	b	c	d	e	f
b	X					
c	X	A				
d	X	X	X			
e	X	X	X	X		
f	X	X	X	X	B	
g	X	X	X	C	X	X



b)

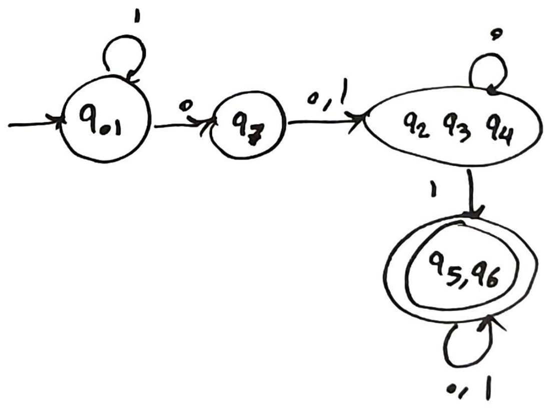
به دلیل وضوح رسم نمودار ادغام نشده است !

$q_2, q_3, q_4 \rightarrow$ معادل

$q_5, q_6 \rightarrow$ accepting + معادل

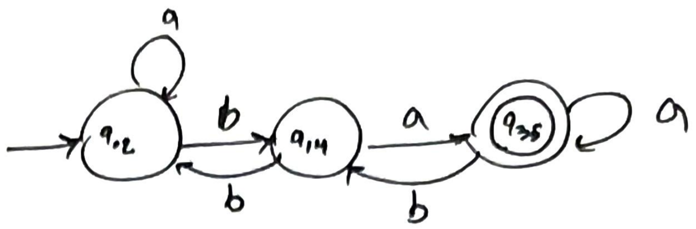
$q_0, q_1 \rightarrow$ معادل

$\Rightarrow$



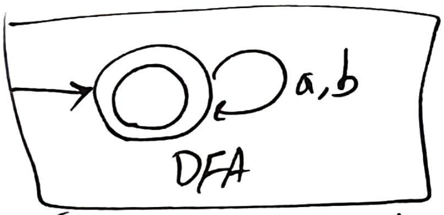
c)

	q_0				
q_1	X				
q_2	A	X			
q_3	X	X	X		
q_4	X	B	X	X	
q_5	X	X	X	C	X



۴)

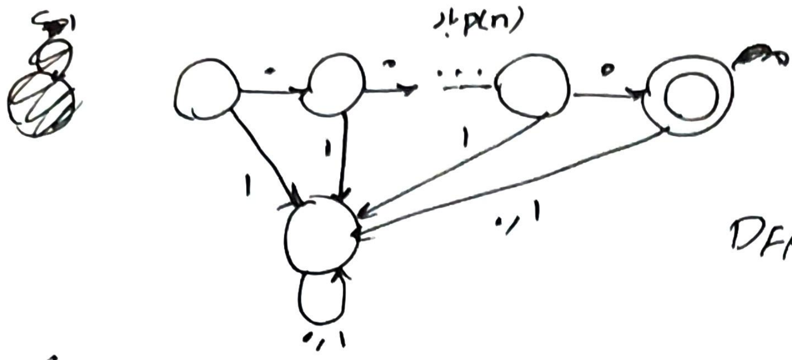
اولین ایراد آن است که اینکه طول w حتما مضرب p است دلیل نمی‌شود که w حتما به صورت a^p باشد بلکه w به صورت a^k است که $k \geq p$ است! با این فرض در مرحله آخر هر توالی را برای w انتخاب کنیم باز نمی‌توانیم برنده بازی باشیم. با کمی تفکر در می‌یابیم زبان منظم است چون DFA زیر بیانگر آن است!



(به طور شهودی می‌توان گفت از سمت چپ حرکت کرده و تعداد a ها و b های سمت چپ (a) و راست (b) cursor را می‌شماریم اختلاف این ۲ مقدار با هر حرکت به سمت راست یا یکی افزایش یا یکی کاهش می‌یابد و با حرکت به سمت چپ کاهش یا افزایش می‌یابد در صورتی که همیشه به سمت کاهش حرکت کنیم حتما یک بار این فاصله صفر می‌شود. با هر حرکت به سمت راست از اول چون ابتدا تعداد a ها صفر و تعداد b ها یک عدد مثبت است یا تعداد a ها را زیاد کرده (a سمت چپ) یا از b های سمت راست می‌کاهیم که در هر صورت در حال حرکت به سمت درست هستیم همچنین بیشترین اختلاف برابر طول string است که با طی کردن کل str به آن می‌رسیم حتما)

(۵)

۱) فرض کنیم p(n) از درجه ۰ باشد یعنی عددی ثابت است و می‌دانیم 0^a یک عبارت منظم است چون DFA آن قابل ترسیم است!

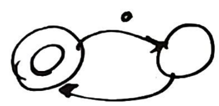


DFA for 0^a $\quad a = \text{const}$

۲) فرض کنیم p(n) از درجه ۱ باشد تعداد صفرهای این عبارت نسبت به n به صورت خطی تغییر می‌کند که در مرحله آخر لم تزریق نیز ما y^i داریم که اینجا نیز توان 0 های موجود در y به صورت خطی رشد می‌کند! در واقع در این حالت داریم:

$$w = 0^{an} = (0^a)^n$$

قبلاً می‌دانستیم که می‌توانیم 0^a را رسم کنیم و برای آن DFA داریم حال چندتا از این DFA را (n تا) پشت سرهم قرار داده‌ایم! همچنین قبلاً در سوالات قبل a^{2n} یا a^{3n} را نیز رسم کرده‌ایم (مثلاً برای تعداد زوج 0 داریم:



)

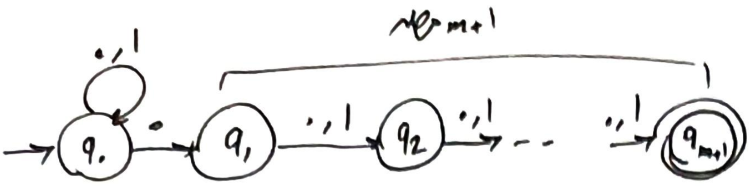
* اما در حالتی که درجه n ۲ نیست چون افزایش به صورت خطی انجام نمی‌شود در انتخاب y های (در مرحله آخر لم تزریق) مختلف شیب پمپش روی تعداد متفاوت شده و شرایط در توان تغییر می‌کند

در واقع اگر در لم تزریق فرض کنیم: $w = xyz \quad x = 0^l \quad y = 0^j \quad z = 0^2$

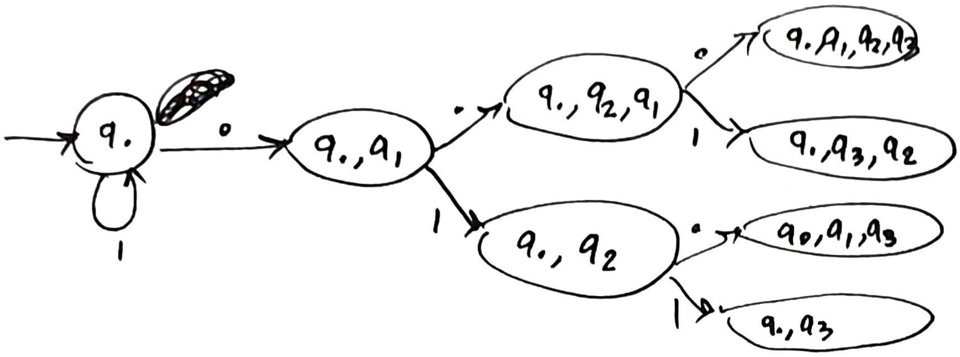
با توان دادن های مختلف y تعداد 0 ها به صورت خطی تغییر می‌کند اما تابع همچنان 0^{an} می‌ماند اما اگر درجه ۲ باشد مثلاً داریم 0^{an^2+bn} با تغییر y در مرحله آخر تعداد 0 ها در حالتی مثل از توان ۲ تغییر کرده و مثلاً خواهیم داشت 0^{an^2+an} که شرایط قبلی حفظ را ندارد و عضو زبان نیست!

(۶)

NFA زبان را می کشیم



سپس NFA رسم شده را باید تبدیل به DFA کنیم.



در شکل بالا تا $m=2$ رسم شده است که ۸ مرحله حاصل! اما چطور؟

در هر مرحله بعد از مرحله‌ی q_0,q_1 یک حالت دوتایی داریم اگر به سمت ۰ حرکت کنیم تعداد state های که در NFA تبدیل به آن state در DFA کمینه شده است یکی افزایش و حرکت به سمت ۱ تعداد آن ثابت است. همچنین چون در NFA جز q_0 که کنار گذاشته‌ایم هیچ self edge نداریم! پس هیچ ۲ state مشابهی در DFA نداریم همچنین برای داشتن یک string با جمله m یک حرف ۰ در ابتدا باعث انتقال ما به حالت شده برای داشتن $(m-1)$ حرف در حالت نیاز به m سطح داریم! پس تمامی حالات یک زیرمجموعه‌ی n تایی ($\forall n \; n\le m$) از مجموعه‌ی m تایی اسامی مراحل را در state های حالت در DFA داریم به جز زیرمجموعه‌ی تهی آن پس از آنجا که تعداد زیرمجموعه‌های هر یک مجموعه‌ی m تایی 2^m-1 است پس تعداد زیرمجموعه‌های این مجموعه‌ی $m+1$ تایی $2^{m+1}-1$ است که با احتساب state اول که q_0 است به تعداد 2^{m+1} می‌رسد!